DATA-DRIVEN ARRAY PROCESSOR

Publication number: CN1051995 (A)
Publication date: 1991-06-05
Inventor(s): SCHMIDT ULRICH [DE]; CAESAR KNUT [DE] +
Applicant(s): SEUTSCHE ITT IND GMBH [DE] +
Classification:
- international: ***G06F15/16; G06F15/167; G06F15/173; G06F15/80;*** **G06F15/16; G06F15/76;** (IPC1-7): G06F15/16
- European: G06F15/173D; G06F15/173N4; G06F15/80A2
Application number: CN19901009263 19901120
Priority number(s): EP19890121506 19891121

Also published as:
EP0428770 (A1)
EP0428770 (B1)
JP3176757 (A)

Abstract not available for CN 1051995 (A)
Abstract of corresponding document: **EP 0428770 (A1)**

A monolithically integratable MIMD (multiple-instruction, multiple-data) array processor (ap) for real-time signal processing consists of a two-dimensional array of cells (zp). The architecture of the cell processors (zp) enables three-address instructions to be executed. Each cell (zp) contains an accumulating multiplier (ma), an arithmetic/logic unit and a multi-port register block (rf). The data traffic within the cell (zp) is handled via a ring and a core bus system. Data are transferred past the cell boundaries by means of a handshake protocol. When the data buffer is full or empty, the cell (zp) transmitting data or receiving data is automatically stopped.

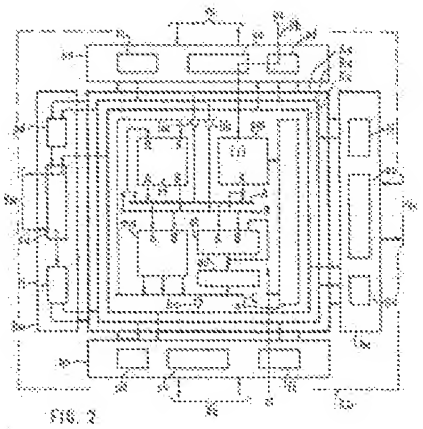


FIG. 2

〔19〕中华人民共和国专利局

〔11〕公开号 CN 1051995A

〔12〕发明专利申请公开说明书

〔21〕 申请号 90109263.0

〔51〕 Int.Cl[5]
G06F 15/16

〔43〕公开日 1991年6月5日

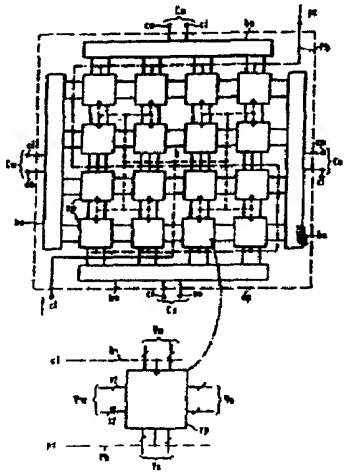
〔22〕申请日 90.11.20
〔30〕优先权
〔32〕89.11.21 〔33〕EP 〔31〕89121506.3
〔71〕申请人 德国ITT工业股份有限公司
地址 联邦德国弗莱堡
〔72〕发明人 乌尔里希·施蒂特
克努特·喀萨尔
〔74〕专利代理机构 中国国际贸易促进委员会专利代理部
代理人 杨晓光

说明书页数: 13 附图页数: 5

〔54〕发明名称 数据驱动阵列处理器

〔57〕摘要

此处揭示的用于实时信号处理的MIMO(=多指令,多数据流)阵列处理器(ap),由单元(zp)的一个两维阵列组成。单元处理器(zp)的体系结构允许执行三地址指令。每一个单元包括一个累积乘法器(ma),一个算术/逻辑单元(al),和一个多端口寄存器单元(rf)。单元(zp)内的数据流通借助于环形总线系统和核心总线系统控制。超出单元范围的数据传递采用握手通信协议方式完成。当数据缓存器充满或空着时,相应的数据发送或数据接受单元(zp)自动截断。



权　利　要　求　书

1．阵列处理器，具有多个相同的单元，它们由同一个时钟信号驱动，位于假想的两维正交网格的结点上，并经由四条通信总线与东西、南、北相邻的单元处理器异步地交换数据，每一个单元处理器包括至少一个算术／逻辑单元（＝ＡＬＵ），一个移位器和一个数据存储器，用于数据处理之目的，其特征在于：

——阵列处理器（ａｐ）的所有单元都集成在一块单独芯片上；

——阵列处理器（ａｐ）是多指令，多数据流处理器（＝ＭＩＭＤ处理器），在其中，每一单元都单独可编程；

——阵列处理器（ａｐ）的四个边缘区域，每一个都包括一个电子总线开关（ｂｓ），它可将一个相邻单元处理器（ｚｐ）的相应通信总线（Ｗｗ，Ｖｏ，Ｖｓ，Ｖｎ），有选择地接到与各边缘区域对应的外部输入和输出端（Ｃｉ，Ｃｏ），通过它们，可同时送入或送出多位数据；

——芯片上的所有单元（ｚｐ）由共同的时钟信号（ｃｌ）驱动，并且

——每一个单元（ｚｐ）含有下列分电路：

——环形总线系统，由一条Ａｘ—源总线（Ａｘ），一条Ｂｘ—源总线（Ｂｘ），和一条Ｃｘ—结果总线（Ｃｘ）组成，至少部分地环绕单元核心；

——两路数据传递装置（＝握手端口）（ｈｗ，ｈｏ，ｈｓ，ｈｎ），它们可在每一个时钟周期内，经由两条数据路径传递新的数据，将环形总线系统连接到东（Ｖｏ），西（Ｖｗ），南（Ｖｓ），

北（Ｖｎ）四条通信总线上，含有供发送数据用的先进先出存储器（＝ＦＩＦＯｓ）（ｆｉ），并且具有阻塞装置，在ＦＩＦＯ空，或满时相应地中断接收或发送单元的信号处理，在这等待状态期间，单元处于冻结状态；

——核心总线系统由Ａ—源总线（Ａ），Ｂ—源总线（Ｂ），和Ｃ—结果总线（Ｃ）构成，並经由Ａ—，Ｂ—，Ｃ—总线寄存器（ｂａ，ｂｂ，ｂｃ）连接到环形总线系统上；

——寄存器单元（ｒｆ）具有连接到核心总线系统上的输入和输出端；

——算术／逻辑单元（＝ＡＬＵ）（ａｌ）（也执行移位和循环功能）其输入端连接到Ａ—和Ｂ—源总线（Ａ，Ｂ）上，且其输出端通过ＡＬＵ延迟器（ａｄ）连接到Ｃ—结果总线（Ｃ）上；

——累积乘法器（＝ＭＡＣ）（ｍａ），其输入端连接到Ａ—，和Ｂ—源总线（Ａ，Ｂ）上，且其输出端连接到Ｃ—结果总线（Ｃ）；

——程序存贮器（Ｐｍ），经由—编程总线（Ｐｂ）（所有的单元（ｓｔ），用于控制单元（２Ｐ）内的数据处理，上述控制单元被装入程序存贮器（Ｐｎ）存贮的数据和单元（２Ｐ）分电路产生的状态信号。

2 权利要求 1 中所述阵列处理器，其特征为：延迟器（ａｄ）的延迟，包括ＡＬＵ（ａｊ）中的传播延迟，等于累积乘法器（ｍａ）中的传播延迟。

3．权利要求 2 中所述阵列处理器，其特征为：在单元（２Ｐ）之内，从数据发送握手端口（ｈｗ，ｈｏ，ｈｓ，ｈｎ）经由环形总

线系统到另一个数据接受握手端口的信号路径延迟等于从一个握手端口，经过单元核心到另一个握手端口的信号延迟，通过使用包括在上述数据接受握手端口之中的延迟器（ｐｄ）达到上述目的。

4。权利要求1中的阵列处理器，其特征为：累积乘法器（＝ＭＡＣ）ｍａ包含有下列分电路：

——并行乘法器，使用流水线作业，并具有通过加法器（ａｄｄ）将它的输出加到累加寄存器，其内容反馈到加法器（ａｄｄ）的另一个输入上。

——加法器（ａｄｄ）进一步提供一个溢出信号（Ｖ）和符号信号（Ｎ），并且

存在累加寄存器中的数据字被分成三个重合的区域，以进行进一步的处理，这三部分有选择地可连接到Ｃ一结果总线（Ｃ）上，分别为

——高位区域（ｈｉ）覆盖最高的位，

——低位区域（Ｌｏ），复盖相邻的低位

——中间区域（ｍｉｄ），复盖贯穿的中间区域的位，上边所说中间区域可在固定的上下限内由限制器（Ｌｉ）可选择地限定。

5。权利要求1中的阵列处理器，其特征为：

——ＡＬＵ（ａｌ）提供一个溢出信号（Ｖ），符号信号（Ｎ），零信号（Ｚ），和一个进位信号（Ｃｒ），以及

——为了级联的功能，ＡＬＵ（ａｌ）的输出数据（Ｄ）直接经由数数据路径，反馈到ＡＬＵ（ａｌ）的两个输入端中的一个。

6。权利要求1中的阵列处理器，特点为：

——可以同时从寄存器单元读出（径由Ａ—输出和Ｂ—输出）和

写入（径由Q—输入或R—输入），

—A—和B—输出相应地接到A—和B—源总线，

—Q—输入既可从A—也可从B—源总线输入，以及

—R—输入从C—结果总线（C）输入。

7．权利要求1中所述阵列处理器，其特征为：存储在程序存储器（pm）中的指令组（i）包括常数（k），通过常数输出（k），即可将它们置于A—也可置于B—总线（A，B）。

8．权利要求1中的阵列处理器，其特征为：程序存储器存有指令组（i）它们的格式由下列各部分构成：

——操作码（oc），

——条件码（sc），它包括作为分支条件的各个状态信号所要求的状态，

——分支地址（bra），

——A—源地址（Aa），用于A—和A_X—源总线（A，A_X）B—源地址（Ba），用于B—和B_X—源总线（B，B_X），以握手端口（h_w，h_o，h_s，h_n）中的一个，寄存器单元（ro，……，R15）中的一个ALU（al），常数（k），或总线寄存器（ba，bb）中的一个做为数据源，

——第一接收点地址（ra），为此目的指令存储器单元（ro，……，r15）中的一个，

——附加接收点地址（Oa，Na，Wa，Sa），它们决定数据传递到相邻的单元，

——C—源地址（Ca），指定了要连接到C—结果总线（C）上的单元核心电路，以及

——寄存器输入地址（Ra），决定寻址了的寄存器单元（ro……，r15）是否必须经由Q一或R一输入端写入。

9．权利要求8中的阵列处理器，其特征为：在指令组（i）中含有常数k，替代条件码（sc），和分支地址（bra）

10．权利要求1中的阵列处理器，其特征为：时钟信号到各个单元（zp）的路由以H树（h）在芯片上布线。

说　　明　　书

数据驱动阵列处理器

在数字信号处理中，例如，在一维或多维的视频信号处理中，阵列处理器越来越受到人们的注意。这些阵列处理器（由通过数据总线相连的多个单元构成）的结构允许实时并行处理信号。这样的阵列处理器如果是时钟驱动的，则称为（Systolic）收缩式阵列处理器；如果是数据驱动的，则称为“波前阵列处理器”。它们在娱乐电子设备中的应用也在增长。在乐电子设备中数字信号处理器的应用越来越广泛。带有无闪烁画面再生的高分辨率电视即为一例。对于电视接收机中的无闪烁画面再生来说，举个例子，在空间的与瞬时的相邻的画面线之间插入附加的行间线是必要的。空间的起始点由被扫描的画面平面定义，瞬时起始点由画面序列给出。

例如，在“计算机”卷20，7号，1987年7月18页—13页中描述了这样的阵列处理器。（题目为“波前阵列处理器——实施概念”）。基于SIMD（＝单指令，多路数据流）原则，每一单元与东西、南、北相邻的单元通信。用握手协议”完成一个单元到另一个单元的数据传递。“握手协议”使得独立的单元与它们各目的相对时钟相位无关地接受数据。由于在独立单元中的数据，并非都是以相同的速度进行计算因此数据源与数据接收点缓存在FIFO（先进先出）存储器之中。对每一个方向上的数据流，在两个相邻单元的数据路径上提供一个FIFO存储器。握手协议通常在一个时钟周期内实现。

在“计算机”卷20，7号，1987年7月，102页至103页中描述了数据驱动阵列处理器。（名为“数据驱动处理器阵列的概念与实施”）上述数据驱动处理器阵列集成在一块VISI（超大规模集成）芯片中，每一个那样的单元可以通过内部环形总线系统与邻近的六个单元交换数据。在阵列处理器之内的几条全程总线确保每一单元能直接与外部计算机通信。

在“IEEE计算机学报”卷C—36，12号，1987年12月，1523页至1538页，描述了一种阵列处理器。（题目为“瓦德全自动远程处理（Warp）计算机体系：结构，实施及性能”）它的单元为，安装在插件板上的处理器模块，能连接到一起以形成一维收缩式矩阵。每一个处理器模块按照MIMD（多指令，多路数据流）原则可以单独编程。因此处理器为完成各式各样的任务提供了高度的灵活性。单个的模块之间通过排队通信。当一个队列（=FIFO）满或空时，相应地将发送或接收模块，阻塞直到队列可以重新处理数据流通为止。那是因为那里已经产生了用于新数据的位置或那已有可用的新数据。这一点使处理器能够极其灵活地偏程，因为序列控制不再需要维护各式各样的单元程序的严格同步。

在“IEEE声学，语言，信号处理国际会议文件汇编”中描述了另外一种MIMD阵列处理器。（题目为“可编程视频信号处理器”1989年2476页到2479页。这种MIMD阵列处理器由集成在一个芯片中的三个单元组成，以双向模式在它们自己之间交换数据，并向芯片外传递数据。这种处理器采用三角形布局，并使用时钟控制单元间通信，能实时处理视频信号。每一个单元具有几个以并行方式操作的处理和存贮器。它们通过纵横制接线器互相联接。每

一部件由“周期静态”程序（周期性执行没有分支的）程序，控制。所有的操作与处理器时钟同步，它的频率是采样频率的整数倍。

欧洲专利申请ＥＰ—Ａ０２７７２６２揭示了一种阵列处理器。这种阵列处理器带有由同一时钟控制的相同的多元单元。这些单元位于一假想的两维格网的结点上，并且经由四条通信总线与东、西、南、北相邻的单元交换数据。单元到单元的数据传递是异步的。每一单元有数据存储器，算术／逻辑单元（ＡＬＵ）和移位寄存器。

本发明的目标是提供一种阵列处理器，它适用于单片集成，允许实时处理从不同信号源而来的数字信号，在外部控制程序（软件）的控制下，它适用于处理大量的不同信号处理任务，特别是用于处理娱电子设备中的视频信号。

现在，参照附图更详尽地说明本发明及其更多的优点。其中：

图１为依据本发明的在一个正方形配置中排列有１６个单元的阵列处理器实施例方框图；

图２为阵列处理器的一个单元的方框图；

图３为带有阻塞装置的两路数据传递装置的部分方框图；

图４为图３的配置的时序图；

图５为累积乘法器（ＭＡＣ）的方框图；

图６为算术逻辑单元（ＡＬＵ）的方框图；

图７为作为程序步骤，要送入的指令组的格式的示意图；

图８ａ和图８ｂ分别表示阵列处理器从线性连接和平面连接时，特殊宽度数据流的接转。

图１方框图中所示的阵列处理器ａｐ含有一个由１６个单元ｚｐ

组成的正方形阵列。每一个单元有一条西方向的通信总线Ｖｗ，一条南方向的通信总线Ｖｓ，一东方向的通信总线Ｖｏ，一条北方向的通信总线Ｖｎ。它们既可连接到相应的相邻单元ｚｐ上，也可连接到四个总线开关ｂｓ中的一个（在外围单元情况下）。每一个总线开关ｂｓ从而汇集了四条外围单元的通信总线。它起到电子多片多位开关的作用。在每一个开关位置，需要连接的通信总线的所有数据输入和数据输出位与相同数量的外部输入和输出端Ｃｉ、Ｃｏ相连接。Ｃｉ Ｃｏ一起组成了外部阵列端口。另外，分配到每一条通信总线上的状态信号（用来执行握手协议）通过总线开关ｂｓ确定路由。由于阵列处理器的四面都配有这样的总线开关，那么在那里也就共有四个外部阵列端口，分别称为西方阵列端口Ｃｗ，南方阵列端口Ｃｓ，东方阵列端口Ｃｏ，北方阵列端口Ｃｎ。

如果，例如，每一通信总线有12个位用于数据输入，12位用于数据输出，那么每一个阵列端口应有12个外部输出端Ｃｏ用于数据输出，12个外部输入端Ｃｉ用于数据输入。在阵列处理器ａｐ以内和以外的数据传递为严格的并行，因此，任意数目的阵列处理器可以连接在一起，以构成使用严格并行数据传递的大阵列。这就可能解决非常复杂的问题，例如，在电视，图像处理，图形学或多维滤波器等领域中。通过以先进的方法组织数据在独立的阵列处理器之内传递，如果阵列处理器以线性模式串行连接（参照图8ａ）则线性数据传递可达到48位。如果它们是以两维模式串行连接（参照图8ｂ）那么在两个独立方向上的数据传输可达24位，这种情况作用于上述假设的例子当中，在那里通信总线有12个输入位和12个输出位。这样就为使用者做出了极其有效的传递装置。两张图中每一张的上半

部分显示出了逻辑数据路径，而下半部分则显示了外部阵列输入和输出端口的对应物理互连。

在阵列处理器中，实时信号的处理需要一个高速时钟，特别是在处理高分辨率电视（ＨＤＴＶ）信号时。为使例如*125*ＭＨｚ的时钟信号同时，分配到芯片上去，应仔细地布设时钟线；否则独立单元ｚｐ相互之间的时钟信号相位差将会变的太大。用于时钟ＣＬ分配的有利方案是所谓的Ｈ树ｈ，它确保分支时钟线均匀一致地装入到它们的终点，将时钟通过一样长度的导线提供给每一个单元。这种配置，例如，在“ＩＥＥＥ计算机学报”，卷Ｃ—*34*，*8*号，*1985*年*8*月*734*页到*740*中有描述，特别是*737*页。文章的题目为“同步大规模ＶＬＳＩ处理器阵列”。在说明本发明的图*1*中，独立单元ｚｐ之间的虚线表示Ｈ树ｈ。

点划线表示编程的总线Ｐｂ，所有的单元ｚｐ都与之相连。经由独立地址送到每一个单元ｚｐ去的单元程序ＰＺ由外部输入。由于在应用程序情况下，单元程序ＰＺ经常保持不变，或很少修改，因此一条串行编程的总线Ｐｂ就足够了。

独立单元ＺＰ也可以按任何四长方形或甚至按一维（即线性）形式配置，不用与图*1*形式一样。单元ＺＰ的数目仅为所用的集成技术所限制。

图*2*显示了单元ＺＰ的方框图。在四个外边的每一边上都有一个两路数据传递装置（握手端口），分别命名为西握手端口ｈｗ，南握手端口ｈｓ，东握手端口ｈｏ，北握手端口ｈｎ。每一握手端口控制数据在相应的通信总线上传递，即西通信总线Ｖｗ，南通信总线Ｖｓ，东通信总线Ｖｏ，北通信总线Ｖｎ。经由与通信总线平行的控制线

（没有在图2中示出），两个握手端口交换用于握手处理的控制信号。以一种握手协议实施。

每一单元ZP的实际信号处理部分（即单元核）由环形总线系统围绕（至少围绕一部分）。环形总线系统由一条Ax—源总线Ax，一条Bx—源总线Bx，一条Cx—结果总线Cx构成，每一条总线为12位总线。每一个握手端口有3个12位数据输入端分别用于Ax—Bx—和Cx—环形总线，以及两个12位数据输出端分别用于Ax—，Bx—环形总线。数据输出由握手端口内的先进先出存储器（FIFO）送入。握手端口的Ax—和Bx—输入与一通道延迟装置Pd相连，它将信号延迟一给定的时间。如同后面将要说明的那样，这一延迟遵循一特定要求。这一要求为，通过环形总线传递的数据，在握手端口上出现的时间不得早于第一个路径单元核的数据。以这种方法，达到外部存取数据的同时性。它独立于在元核中的各目的处理。从单元核而来的数据送到Cx—结果总线C_x上。从这里将此数据传递到一个握手端口。由于这个数据不需要再延迟，它被直接传递到位于握手端口中的序贯电路SU上。实施了这一握手通信协议，序贯电路SU即可以传递从A_x或B_x—源总线来的延迟了的信号，也可以来自C_x—结果总线的未延迟信号经通信总线传递到相邻的单元。将收到的信号通过相应的通信总线，按照握手通信协议传递到序贯电路SU，并且装入到FIFOfi中进行缓存。从这里，如上所述，它即可送到A_x—，也可送到B_x—源总线。

单元ZP中的控制单元。按照存储在程序存储器Pm中的指令组i控制上述及进一步的操作。为清楚起见，图2中仅显示了数据链，它们经常以多位总线的形式实施，而没有给出控制线。由于所有的握

手端口是一样的，这里仅给出西握手端口 h w 中内部数据路径的细节。

环形总线系统允许一个单元内不同握手端口之间非常灵活地交换数据。例如，可以在环形总线 A x，B x，C x 上传递三个独立的数据流，源总线 A x，B x 正在从各一端输入，而结果总线 C x 可以一次输入到四个端口。

除了环形总线系统之外，单元 Z P 包括一个用于内部信号处理目的的核心总线系统。这一总线系统由 A —源总线 A，B —源总线 B，及 C —结果总线 C 构成。A x —源总线和 B x —源总线经由 A —总线寄存器 b a 和 B —总线寄存器 b b 向相应的 A —和 B —源总线 A，B 输入。同样地，C —结果总线 C 经由 C —总线寄存器 b c 向 C x —结果总线 C x 输入。这些总线寄存器将核心总线系统与环形总线系统解耦，并且还允许数据字存在其中，直到新的数据字将它们复盖。在由另一个访问握手端口读操作的复盖那些数据字之前，单元核心的信号处理电路都可以使用那些由握手端口读出的并暂存在总线寄存器 b a，b b 中的数据字。

在单元核心的信号处理由累积乘法器（＝M A C）m a 和算术／逻辑单元（＝A L U）a l 运算。累积乘法器的输入端连接到 A —和 B —源总线 A，B 其输出端连接到 C —结果总线 C。算术／逻辑单元（＝A L U）还具有移位及循环移位功能，它与通向 A —和 B —源总线 A，B 相接，并且其输出端与经由 A L U 延迟单元 a d 与 C —结果总线 C 相接。

快速数据缓存发生在寄存器单元 r f 之中，那里有，例如，*16* 个可选择存取寄存器单元 r o，……，r *15*。对于快速存取，这个

寄存器可同时通过A—输出和B—输出读出及通过Q—输入或R—输入写入。因此，寄存器单元rf适用于单元中的三地址处理，它在时钟C1的每一周期内组合两个操作数并存储一个结果。寄存器单元rf的R—输入仅由C—结果总线C装入，而Q—输入既可由A—源总线A装入，也可由B—源总线B装入。

由于MAC ma的流水线深度大于ALUal的流水线深度，因此ALU延迟期ad的延迟产生了时间补偿。对于那些使用以前的操作结果作为输入的操作来说——这些也被称作级联操作——ALUal的输出D被直接反馈到ALU的一个输入。在图2中，这是AALUal的输入。使用这种直接反馈路径，最大限度地减小了ALU延迟期ad的等待时间。

存储在程序存储器Pm中的指令组i也可含有常数k，它由常数输出k输出到A—源总线A或B—源总线B。

图3显示了两路数据传递装置的部分方框图。如上所述它也被称作握手端口。为了清楚起见，仅显示出数据在一个方向上传输所必需的那一部分电路。对于双向数据传递，每一个握手端口含有数据发送器Se和数据接受器em。必须认为门tr是源序贯电路SU1的一个组成部分。它缓存了数据字dat。此数据字是要经由数据总线ddb被传递到数据接受器em的。

在数据接收器em中的接收握手端口包括FIFO存储器fi和接收点序贯电路SU2。数据发送器Se包括数据源dq，它由第一时钟信号CL1计时，并且可以做为单元核中的一个数据源，（仅作例子）。在数据接收器em中相应的装置为数据接收点ds，它由第二个时钟信号CL2控制。并且是，例如，在接收单元核中的一个数

据接收点。两个时钟ＣＬ*1*，ＣＬ*2*，具有同样的频率，但是它们的相位可能由于不同的延迟而不同，参照图*4*。

即使由于两个时钟延迟不同的时间量，或即使它们频率不一样，上述异步握手通信协议也能保证正确的数据传递。现在参照时序图，图*4*，来说明操作序列。

如果数据发送器ｓｅ要发送一个数据字ｄａｔ，它要在时刻*1*时向接收点序贯电路ＳＵ*2*送出一个请求信号ｒｅｑ。只要在从存储器序贯电路而来的确认信号ａｃｋ没有置位时，才可送出请求信号。一确认信号没有置位，表明数据接受器ｅｍ已准备好接受一个数据字。一数据字ｄａｔ的接受是在时刻*2*，通过置位一个确认信号ａｃｋ，向源序贯电路ＳＵ*1*发出信号的。在信号字ｄａｔ被真正接受前，不发送这一信号。数据信号ｄａｔ必须在请求信号ｒｅｑ开始的达到它的稳定状态。

源序贯电路ＳＵ*1*通过在时刻*3*时复位请求信号ｒｅｑ来确认收到了确认信号。同时，也可取消数据字*3*。在时刻*4*时，通过复位确认信号ａｃｋ向源序贯电路ＳＵ*1*发出信号表示接收点序贯电路ＳＵ*2*，已经做好接受新数据字ｄａｔ的准备。这表明在ＦＩＦＯ存储器ｆｉ中至少有一个单元是空的。新的数据传送最早可在时刻*5*启动。这样，整个字的传递可在一个时钟周期内以异步的方式进行。

请求信号以及确认信号的发出和取消是与从数据发送器ｓｅ和数据接收器ｅｍ来的状态信号逻辑地组合在一起的。例如，仅在源状态信号ｓｑ发出表明数据源ｄｑ已经产生了一个新的数据字ｄａｔ的信号时，才发出请求信号ｒｅｑ。在接收方，确认信号ａｃｋ的置位跟在请求信号ｒｅｑ置位以后。如果第一接收点状态信号ｓｖ表明在

ＦＩＦＯ存储器ｆｉ中至少有一个单元有空，则确认信号才被取消。在数据发送器ｓｅ中确认信号的取消之后。取消请求信号ｒｅｑ的。如果没有取消确认信号的话，源序贯电路ＳＵ*1*不能启动传递新的数据字。

为使数据传递在一个时钟期内能够完成，握手通信协议的所有四个阶段必须在一个时钟周期内实施。这一点可利用现有的任何高频率时钟以异步方式或以同步方式完成。通过锁定握手通信协议的单个阶段，在任何情况下均能正确地传递数据。

很明显，就平均数而言，如果数据接收器ｅｍ接受的数据与数据发送器ｓｅ产生的数据一样多，那么大规模ＦＩＦＯ存储器ｆｉ就能进行大量的独立信号交换。如果这一点不再保证，那么既有可能丢掉在数据源ｄｑ中的数据，也有可能数据接收点ｄｓ从空的ＦＩＦＯ存储器ｆｉ单元中读数。通过插入ｎｏｐ（空操作）指令可以在程序中避免这种必须避免的情况。然而，这就要求精确地监视在阵列处理器中单个的信号路径的运行次数，并且使得编制程序极其困难。

因此，依据本发明的一个特殊的优点，序贯电路ＳＵ*1*，ＳＵ*2*自动地保证了程序员不必照看单个数据路径的逻辑同步。例如，从源序贯电路ＳＵ*1*来的源截断信号Ｓｔ*1*将（如果确认信号ａｃｋ发出数据接收器ｅｍ没有准备好接受数据的信号时，将截断信号ｓｔ，发出）截断数据源ｄｑ和门ｔｒ。另一方面，如果从ＦＩＦＯ存储器ｆｉ来的第二状态信号ｓＬ表示那里所有的单元都是空的，那么在数据接受器ｅｍ中的数据存储器ｄｓ将被从存储器序贯电路ＳＵ*2*来的接收点截断信号ｓｔ*2*阻塞住。只要截断还有效，相应单元的状态保持“冻结”状态；更具体地数据之间的时间关系将被保持住。单元状

态的“冻结”需要增加单元zp中电路的数量，然而，由于增加了编程便利性使之具有充分的理由来这样做。

使用上述的握手控制，数据可 步地从握手端口传递到握手端口口。握手通信协议的暂时实施在细节上是高度可变的，因此即使是非常大的不同延迟——例如——越过芯片边界的——都能考虑进行。

图5中方框图所示累积乘法器（＝MAC）由一个并行乘法器mp构成，它的A—和B—输入，每一个，例如，具有12位。并行乘法器mp的带符号的输出信号被加到加法器add的一个输入端，它的另外一个输入由累加寄存器ar（它的输入与加法器add的输出相连接）的输出信号提供。在图5ho实施例中，累加寄存器ar有29位的存储器容量。累加寄存器ar的五个附加位代表在二个12位数相乘时最大可能的累加范围。在二进制补码表示时，它们也包括符号位。

加法器add还传递两个附加信号，一个是溢出信号V，它表明运算超出了事先定好的数的范围，另一个是符号信号N，它表明相加的结果为负数。

可将累加寄存器的内容以三种不同方式装入C—结果总线C。例如，如果C—结果总线的宽度仅有12位，那么，当然在这条总线上最多只能装入12位。一种可能的方法是相继地读出寄存器内容，即第一次读出最高的12位作为高位区域hi，然后紧接着的12个低位作为低位区域Lo。对累加寄存器中最低的五位不加考虑。第二种可能的方法为从中间区域里读出12位，例如从占据寄存器11，到22位中读出。如果相乘的结果基本保存在这一中间区域mid中那么对这一区域进行进一步的处理是适当的。这一点也应用于数值从

－1到＋1范围之间的定点操作数乘法运算。然而，任何中间范围的剩余数，即使这个中间区域mid通过一个极限器Li，极限器Li可以通过程序激活并保存数值范围的上限和下限数据法，也不会使结果产生干扰性跳跃。

图6的方框图，表示了算术／逻辑单元（＝ALU）aL。它的两个12位输入A，B分别连接到A—源总线，和B—源总线。数据输出D也有12位，它提供ALU的结果daL，并反馈到第二ALU输入。使用这一数据反馈，可以进行包括进位信号在内的（如果需要）级联移位和循环功能除去传递ALU结果daL，ALU还作为附加状态信号。提供下列状态信号：溢出信号V，在溢出发生时，符号信号N，在结果为负时；零信号Z，在结果为零时；和进位信号Cr。

图7，图示了指令组i的格式（它含有，例如48位）。作为一个程序步骤，将它输入。第一个区域含有用于控制单元st的编码指令作为操作码OC。第二个区域含有一条件码se，它写入单元核及握手通信端口的状态信号。第三个区域含有一分支地址bra，它依照状态码sc和单元核及握手通信协议的现行状态指定含在程序存储器pm中的程序序列。这两个区域，（它们至少含有12位）也可以不用来存储条件码sc和分支地址bra，而用来存一常数k，它通过常数输出k，如上所述，装到A—或B—源总线A，B。在第四和第五个区域，分别确定环形总线系统的A—和B—源总线地址Aa，Ba。经过一个时钟周期的延迟，这些地址也应用于核心总线系统。用于这一目的数据源是，例如，握手通信端口hn，ho，hs，hn中的一个，寄存器单元ro，……r15，中的一个，ALUal，

常数k，或总线寄存器ba，bb中的一个。这个源的定义后跟着5个区域，它们决定了数据向何处装入。因此，它们含有接收点地址。在第六个区域，第一接收点地址ra指明，C—结果总线C必须向寄存器单元ro，……，r15中的哪一个装入。第七，第八，第九和第十个区域含有第二接收点地址，它们为相应地以要选择的握手端口地址形式Oa，Na，Wa，Sa表示，它们决定了数据传送到相邻的单元。

每一个这样的区域含有两位，以指明，数据从环形总线系统中的三条总线中的哪一条之中产生，或端口是否保持“沉默”，即完全不传送数据。此种状态与空操作指令一致。

第十一个区域含有一个C—源地址Ca，它决定了要联接到C—结果总线C上的单元—核心电路。第十二个区域含有寄存器输入地址Ra，它决定了在第六个区域中寻址了的寄存器单元ro，……，r15是经由Q—还是R—输入写入。

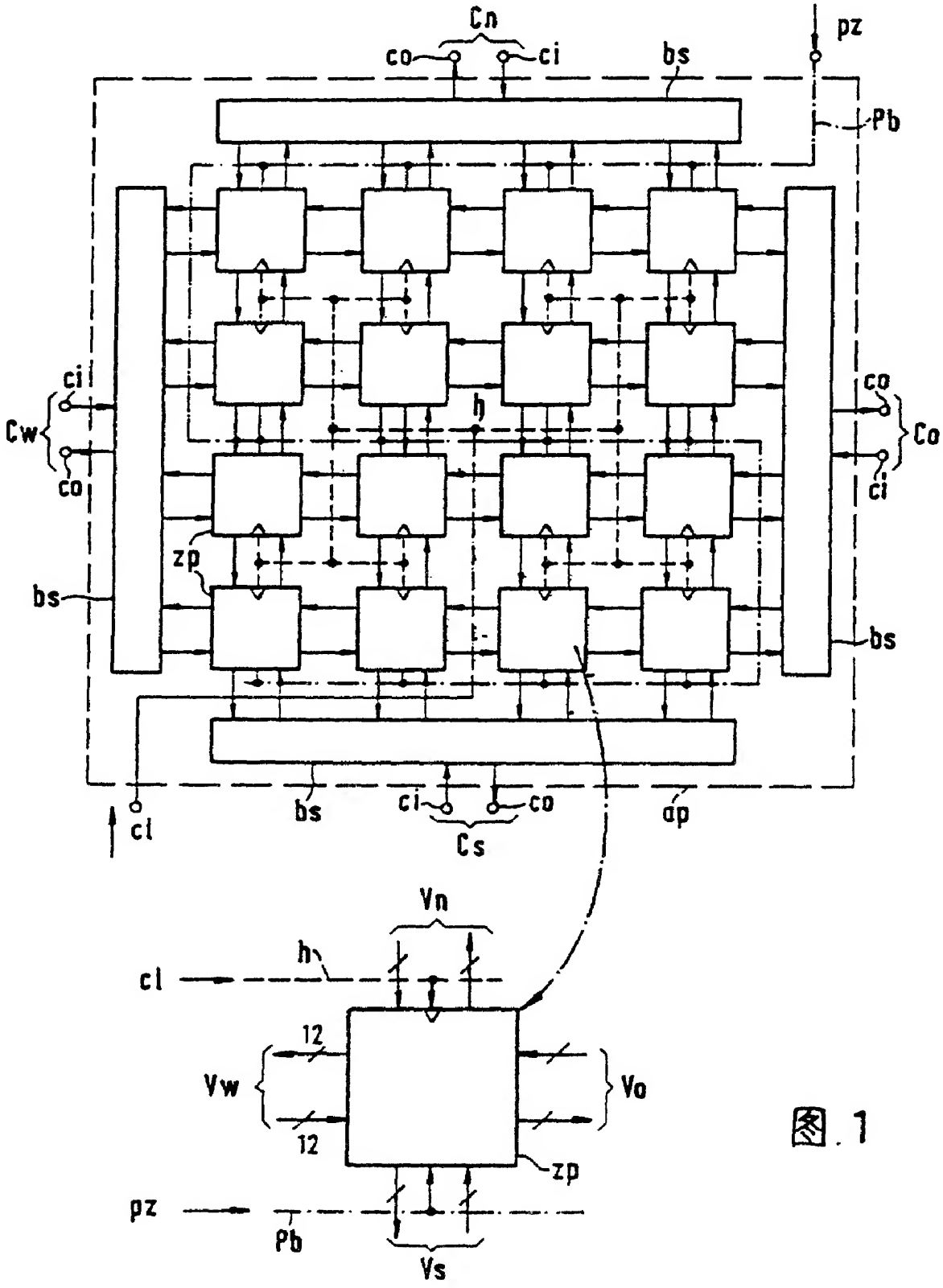

Cn
co
ci
bs
pz
Pb
Cw
ci
co
h
Co
co
ci
zp
bs
bs
cl
bs
ci
co
Cs
dp
Vn
h
cl
12
Vw
12
Vo
zp
pz
Pb
Vs

图. 1

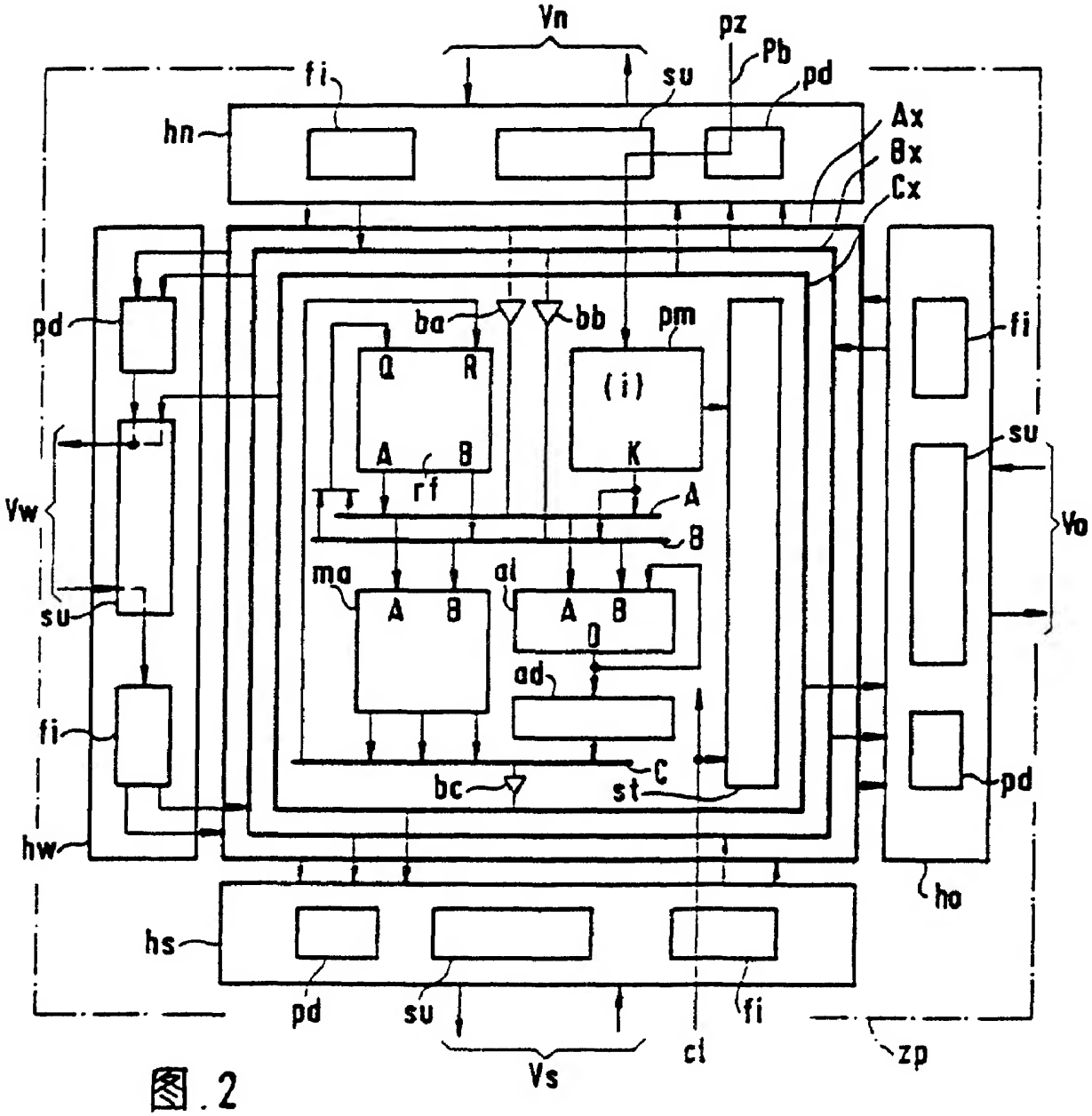

Vn
pz
Pb
fi
su
pd
hn
Ax
Bx
Cx
ba
bb
pm
Q
R
(i)
A
B
K
rf
A
B
Vw
ma
al
A
B
A
B
D
su
ad
fi
C
bc
st
hw
fi
su
Vo
pd
ho
hs
pd
su
fi
Vs
cl
zp

图 . 2

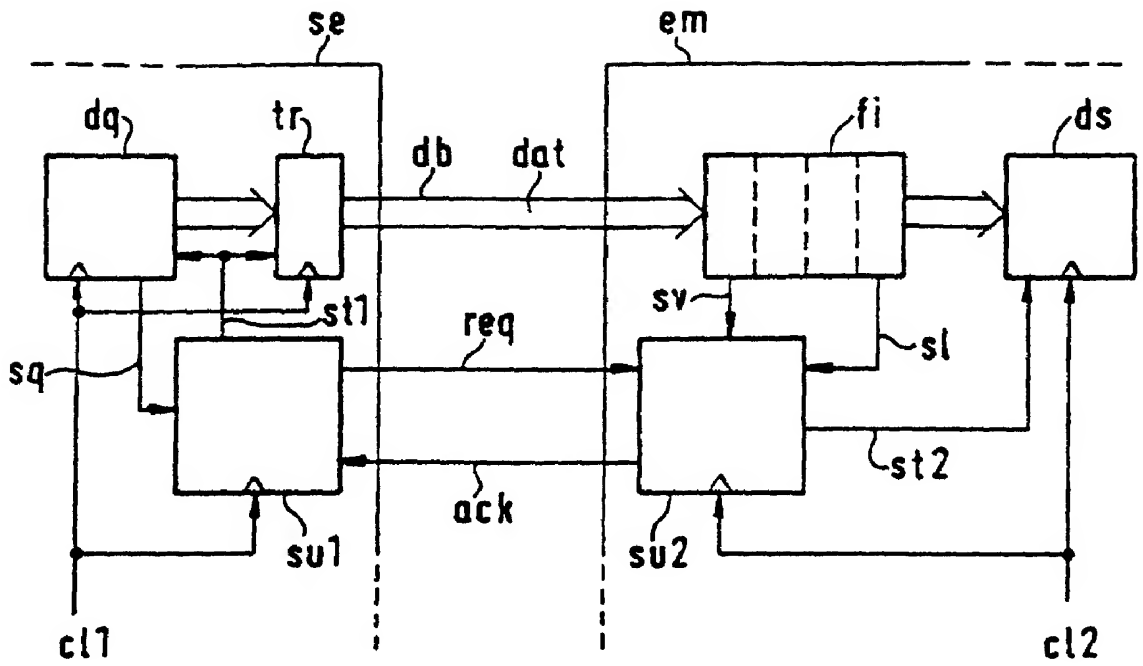

se
em
dq
tr
db
dat
fi
ds
st1
req
sv
sq
sl
st2
ack
su1
su2
cl1
cl2

图.3

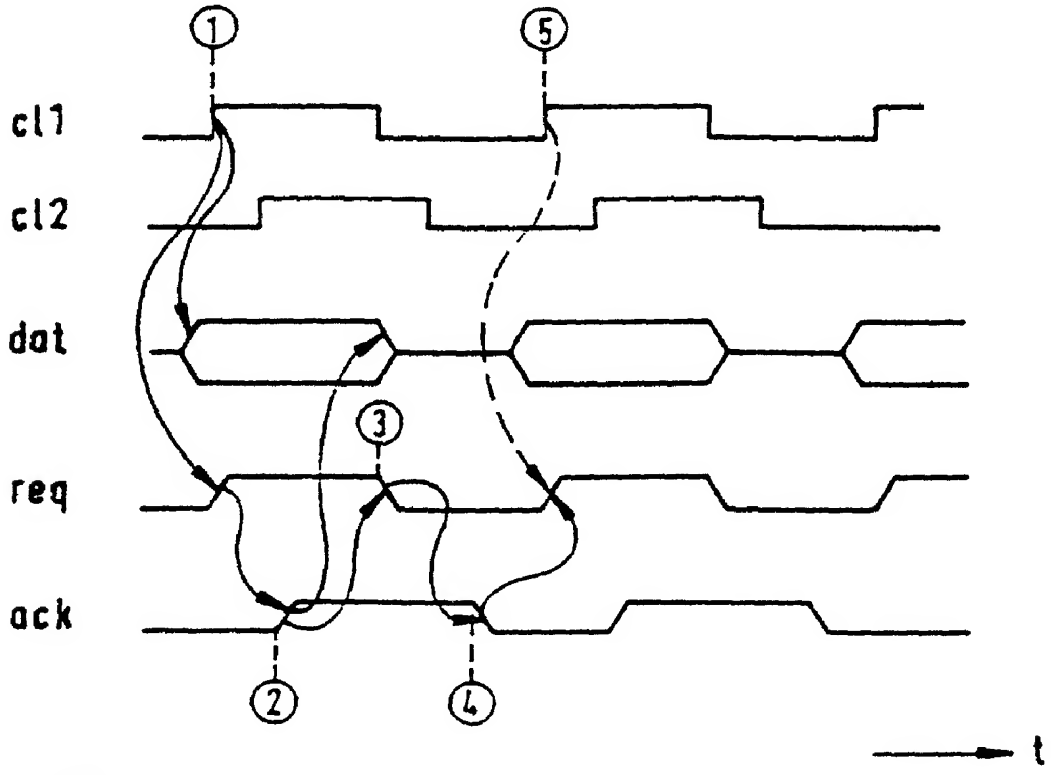

1
5
cl1
cl2
dat
3
req
ack
2
4
t

图.4

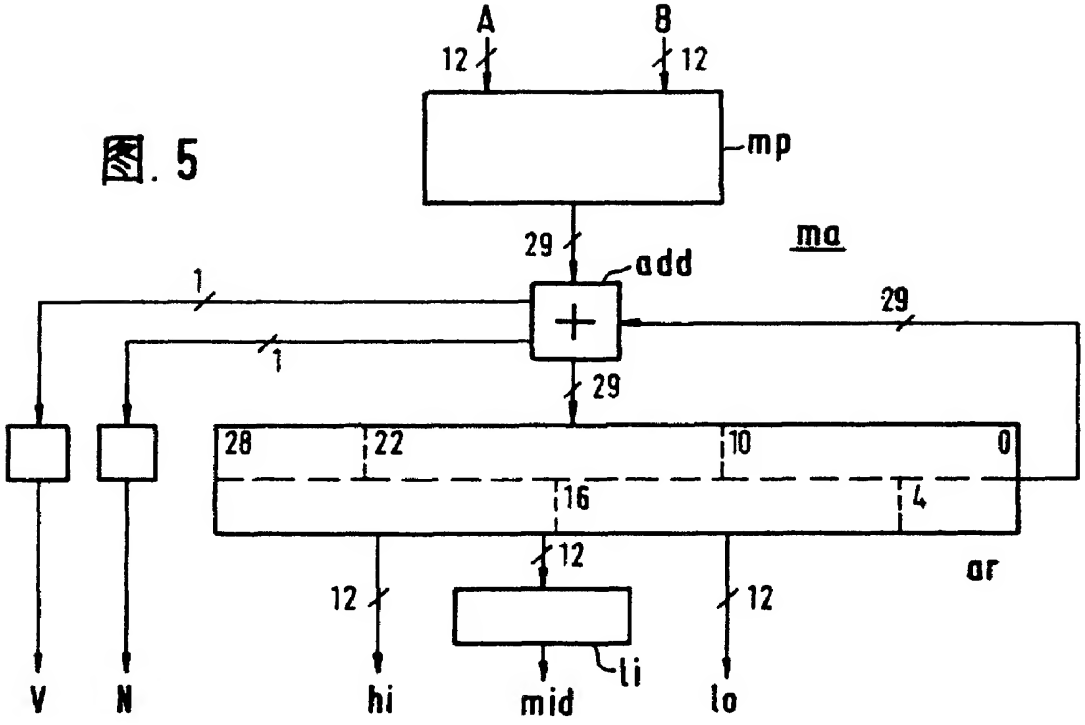

A
B
12
12
mp
29
ma
add
1
1
29
29
28
22
10
0
16
4
12
ar
12
12
V
N
hi
mid
li
lo

图. 5

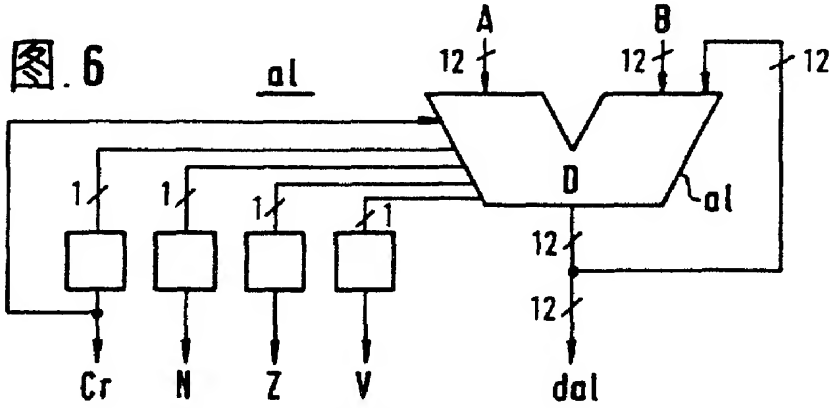

A
B
12
12
12
al
D
al
1
1
1
1
12
12
Cr
N
Z
V
dal

图. 6

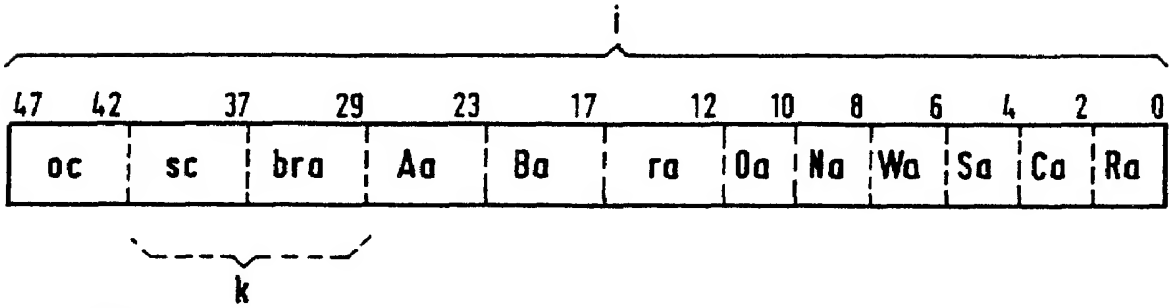

i
47
42
37
29
23
17
12
10
8
6
4
2
0
oc
sc
bra
Aa
Ba
ra
Oa
Na
Wa
Sa
Ca
Ra
k

图 7

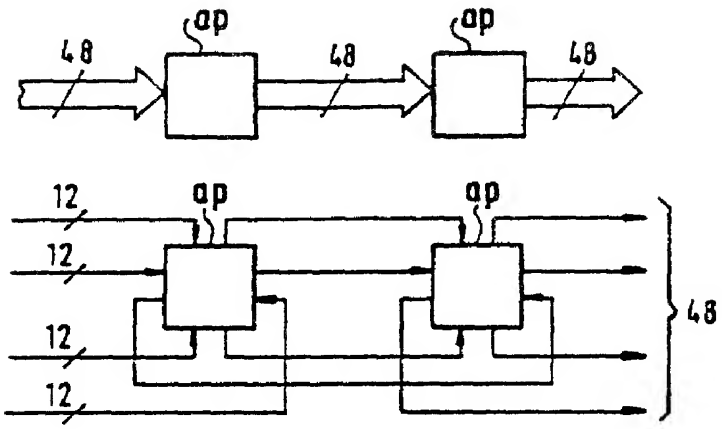

ap
ap
48
48
48
12
12
12
12
ap
ap
48

图.8a

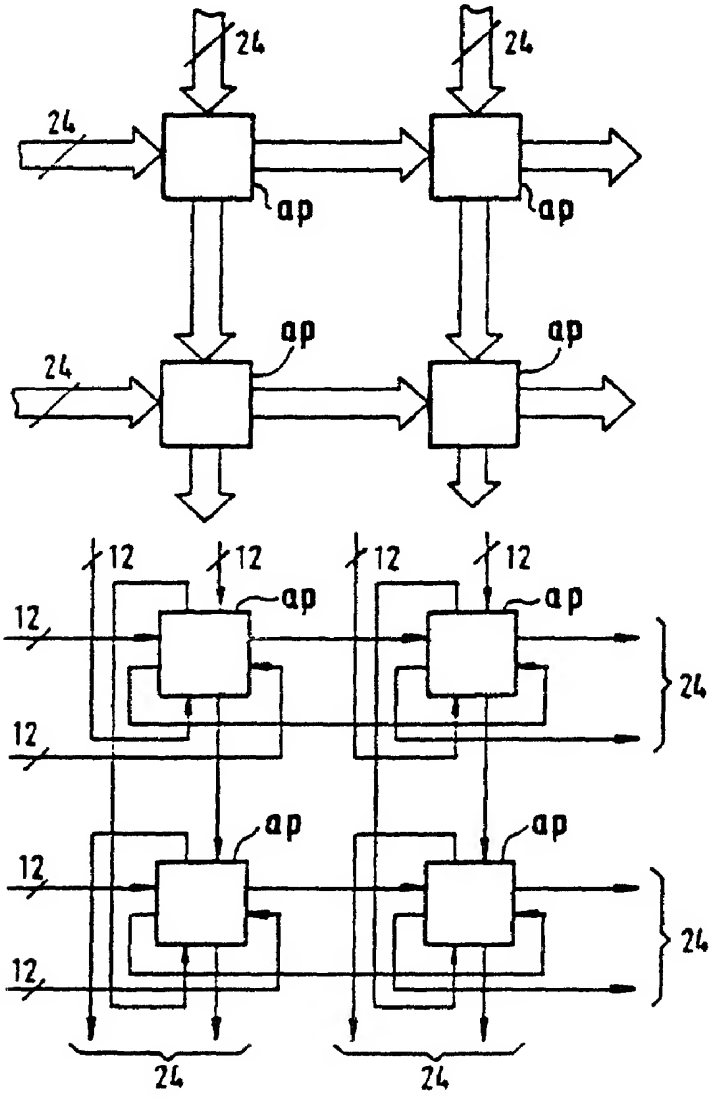

24
24
24
ap
ap
ap
ap
24
12
12
12
12
ap
ap
12
12
ap
ap
12
12
24
24
24

图.8b